
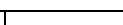


**SCIENCE TECH: Jurnal Ilmu Pengetahuan dan Teknologi**

	Volume 7, No. 1, Bulan Februari, hal. 26-34		
	science.tech@ustjogja.ac.id		
	ISSN	ISSN	
	2460-6286 (Print)	2579-3624 (Online)	

**Analisis Perbandingan Performa Virtualisasi Berbasis Container  
dengan Virtualisasi Berbasis Hypervisor**

***Comparative Analysis of the Performance of Container  
Based Virtualization and Hypervisor Based Virtualization***

**Stefanus Eko Prasetyo<sup>1</sup>, Benny<sup>2</sup>**

Fakultas Ilmu Komputer, Universitas Internasional Batam<sup>1</sup>  
*stefanus@uib.ac.id<sup>1</sup>*

Fakultas Ilmu Komputer, Universitas Internasional Batam<sup>2</sup>  
*bennyzhangz15@gmail.com<sup>2</sup>*

<b>Info Artikel</b>	<b>Abstract</b>
<b>Naskah diterima:</b> 08/01/2021	<p><i>Usage of virtualization for company is nothing new in this era. Virtualization helps company in reducing the resource they needed for creating new different servers. Among the virtualization technology that exists, there are two virtualization technologies that dominated the market, Hypervisor and container. Hypervisor is a virtualization technology which allow machine to create different virtual machines with different operating system, and container is a virtualization technology which isolated in operating system level. With different architecture, the performances between the two virtualization are considered different too. Two identical machines are created in cloud with performance test ran on both machines using apache jmeter to find out the performance between the two virtualization technology. The key variable in this performance test are response time, deviation, throughput, error, and CPU Utilization, with two different setting in each test, the first with 200 users and the second with 500 users. In the load of 200 users, machine using Hypervisor virtualization technology performs the same as the machine using container virtualization technology. However at load of 500 users, machine using Hypervisor virtualization technology fails to keep up and lose a significant amount of performances than the machine using container virtualization technology.</i></p>
<b>Naskah direvisi:</b> 12/01/2021	
<b>Naskah disetujui:</b> 19/01/2021	
<b>Korespondensi Penulis:</b> <i>bennyzhangz15@gmail.com</i>	<p><b>Keywords:</b> Virtualization; Hypervisor; Container; Performance Test.</p>
<b>m</b>	<b>Abstrak</b>

---

Penggunaan virtualisasi bagi perusahaan merupakan sesuatu yang tidak asing lagi. Virtualisasi membantu perusahaan untuk menghemat sumber daya yang diperlukan dalam membangun server-server yang berbeda. Diantara virtualisasi yang ada, terdapat dua jenis virtualisasi yang mendominasi, yaitu *Hypervisor* dan *container*. *Hypervisor* adalah virtualisasi memungkinkan suatu mesin untuk menjalankan beberapa mesin virtual dengan sistem operasi yang berbeda, dan *container* adalah virtualisasi yang mengisolasi pada level sistem operasi. Terdiri dari arsitektur yang berbeda, performa dari kedua virtualisasi juga dianggap berbeda. Untuk menemukan perbandingan performa dari kedua virtualisasi maka dibangun dua mesin berbasis *cloud* yang memiliki spesifikasi yang sama kemudian dijalankan performance test pada kedua mesin tersebut dengan menggunakan apache jmeter. Variabel yang menjadi acuan pada performance test ini adalah *response time*, *deviation*, *throughput*, *error* dan *CPU utilization*, dengan dua jenis pengujian yang pertama dengan 200 *user* dan yang kedua dengan 500 *user*. Mesin yang menggunakan virtualisasi berbasis *Hypervisor* menghasilkan performa yang sama dengan mesin yang menggunakan virtualisasi berbasis *container* pada 200 *user*, namun memiliki penurunan performa yang signifikan ketika dihadapi dengan 500 *user* dibandingkan dengan mesin yang menggunakan virtualisasi berbasis *container*.

---

**Kata Kunci:** Virtualisasi; *Hypervisor*; *Container*; *Performance Test*.

---

**Sitasi:** Stefanus Eko, P., Benny. (2021). Analisis Perbandingan Performa Virtualisasi Berbasis *Container* dengan Virtualisasi Berbasis *Hypervisor*. Science Tech: Jurnal Ilmu Pengetahuan dan Teknologi. Volume: 7, No. 1, Februari, hal. 26-34.

---

## Pendahuluan

Seiring dengan perkembangan zaman, teknologi IT yang ada pada dunia ini juga semakin berkembang. Untuk mengaktraksi pengguna zaman sekarang yang lebih mengutamakan *cost-efficient*, developer mengeluarkan berbagai teknologi yang bertujuan untuk menghemat sumber daya. Virtualisasi merupakan salah satu teknologi yang berada pada kategori tersebut. Virtualisasi merupakan istilah untuk sebuah teknologi yang digunakan untuk membuat sebuah sistem virtual yang berfungsi seperti sistem yang asli (Widarma & Siregar, 2019).

Terdapat dua macam teknik virtualisasi yang menjadi pilihan banyak orang dalam mengaplikasikan teknologi virtualisasi, yaitu virtualisasi berbasis *hypervisor* dan virtualisasi berbasis *container*. *Hypervisor* merupakan teknologi virtualisasi yang menjadi landasan agar berbagai sistem operasi dapat berjalan secara bersamaan pada sebuah mesin (Kurniawan & Kurniawan, 2016). Sedangkan *container* merupakan suatu teknologi virtualisasi yang melakukan isolasi pada tingkat sistem operasi (Potdar, Narayan, Kengond, & Mulla, 2020).

Salah satu aplikasi yang menawarkan teknologi virtualisasi berbasis *container* adalah *docker*. Secara teori, *docker* yang menawarkan virtualisasi berbasis *container* dianggap lebih

bagus performanya dibandingkan virtualisasi yang berbasis *hypervisor* seperti *virtual box* karena lebih efisien berdasarkan arsitektur teknologinya (Rad, Bhatti, & Ahmadi, 2017).

Menurut Kumar & Kurhekar (2017) dengan penelitian yang berjudul “*Economically Efficient Virtualization Over Cloud Using Docker Containers*”. *Container* memiliki tingkat skabilitas yang besar, aman, mudah digunakan dan dikembangkan dibandingkan dengan teknik virtualisasi pendahulunya sehingga lebih *cost-efficient*. *Container* menghasilkan performa yang bagus dan menggunakan sumber daya yang lebih minimal.

Menurut hasil penelitian (Morabito, Kjällman, & Komu, (2015), performa dari virtualisasi berbasis *hypervisor* telah berkembang banyak dari tahun ke tahun walaupun masih ada kekurangan seperti dalam Disk I/O efficiency. Performa *container* dipastikan tetap lebih baik dan memiliki keamanan yang lebih bagus. Namun, melihat perkembangan *hypervisor* yang meningkatkan performanya dari tahun ke tahun tidak menutup kemungkinan teknologi *hypervisor* yang baru bisa menghasilkan performa yang lebih baik dari *container*.

Penelitian yang dilakukan oleh Li, Kihl, Lu, & Andersson, (2017), juga menunjukkan rata-rata performa yang dihasilkan oleh *container* secara umum lebih baik dari pada *hypervisor* hingga dapat dibandingkan langsung dengan mesin fisiknya. Namun berdasarkan penelitian yang dilakukan oleh Raho, Spyridakis, Paolino, & Raho, (2015) dan Wang, Song, Cui, & Cao, (2017) menunjukkan bawa perbedaan performa antara kedua jenis virtualisasi bisa bervariasi tergantung dari jenis pekerjaan yang ditugaskan kepada masing-masing virtualisasi dan tidak menemukan perbedaan performa yang besar antara kedua virtualisasi.

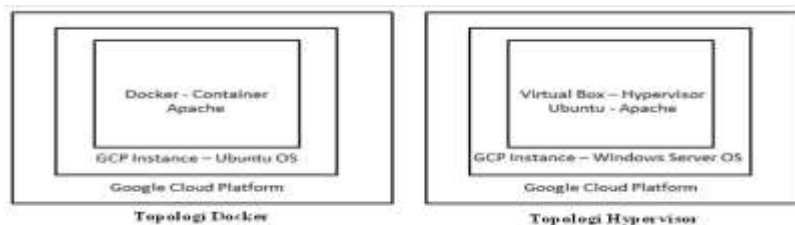
Berdasarkan teori penelitian yang telah dilakukan terlebih dahulu oleh para peneliti, maka dirancang sebuah penelitian untuk menganalisis kembali perbandingan performa antara virtualisasi berbasis *container* dengan virtualisasi berbasis *hypervisor* untuk mengetahui bagaimana performa dari masing-masing virtualisasi jika di uji menggunakan Apache Jmeter.

## Metode

Penelitian ini dilakukan dengan metode eksperimental, dilakukan pengukuran terhadap variabel tertentu dalam perlakuan yang berbeda. Variabel-variabel yang digunakan dalam penelitian ini adalah:

1. *Response time*, waktu yang dibutuhkan server untuk merespon request klien. Terdiri dari min, max dan average.
2. *Deviation*, perbedaan *response time* dari tiap user dengan average response time yang ada.
3. *Throughput*, jumlah request yang bisa diproses oleh server dalam jangka waktu tertentu.
4. *Error*, jumlah terjadinya kesalahan atau gagal respon dari server.
5. *CPU Utilization*, jumlah sumber daya CPU yang digunakan oleh server ketika diakses oleh klien.

Kedua server diberikan dua perlakuan, yang pertama dalam load 200 user dengan loop sebanyak 20 kali, yang kedua dalam load 500 user dengan loop sebanyak 20 kali. Dalam pelaksanaan penelitian ini dibangun dua mesin dengan spesifikasi yang sama menggunakan *google cloud platform* sebagai *platform* penyedia layanan *cloud*. Desain sistem pada masing-masing virtual dapat dilihat pada Gambar 1.



Gambar 1. Topologi Sistem

Beberapa perangkat keras yang digunakan dalam penelitian ini adalah sebagai berikut:

1. Sebuah *cloud server* untuk tempat pengimplementasian *instance docker* dan *virtual box*. Dengan masing-masing instance memiliki spesifikasi utama:  
 Processor : 2 Core  
 Memory : 8 GB  
 Harddisk : 50 GB
2. Sebuah laptop untuk melakukan analisa performa pada *docker & container*.  
 Beberapa perangkat lunak yang digunakan dalam penelitian ini adalah sebagai berikut:
  - a. *Docker* version 19.03.13 sebagai virtualisasi berbasis *container*, menggunakan image *httpd* (*apache2*) version 2.4.46.
  - b. *Virtual Box* version 6.1.16 sebagai virtualisasi berbasis *hypervisor*, menggunakan sistem operasi *Ubuntu 16.04.7 LTS*.
  - c. *Apache2* version 2.4.46 sebagai aplikasi yang akan diimplementasikan pada kedua virtualisasi.
  - d. *Apache Jmeter* version 5.3 sebagai alat untuk menganalisa performa *web server* yang terpasang pada virtualisasi masing-masing.

## Hasil dan Pembahasan

### Hasil

Pengujian dilakukan terhadap kedua server yang sudah terbangun dalam *google cloud platform* dengan masing-masing server menggunakan virtualisasi yang berbeda. Yang satu menggunakan *hypervisor* dan yang satunya lagi menggunakan *container*. Instalasi *apache web server* dilakukan kepada masing-masing server juga sebagai aplikasi dimana *performance test* akan dijalankan dengan menggunakan *apache jmeter*. Data yang didapat disusun berdasarkan masing-masing variabel yang digunakan pada penelitian ini.

**Tabel 1.** Hasil *Response Time*

Server	Response Time					
	200 User			500 User		
	Min	Max	Avg	Min	Max	Avg
<i>Docker</i>	34	211	60	36	1184	93
VB	34	248	72	42	18435	553

**Tabel 2.** Hasil *Deviation*

Server	Deviation	
	200 User	500 User
<i>Docker</i>	19.75	122.39
VB	20.39	1276.81

**Tabel 3.** Hasil *Throughput*

Server	Deviation	
	200 User	500 User
<i>Docker</i>	324.3	329.9
VB	322.8	282.8

**Tabel 4.** Hasil *Error*

Server	Error					
	Total Req.	200 User Comp. Req.	Error (%)	Total Req.	500 User Comp. Req.	Error (%)
<i>Docker</i>	4000	4000	0	10000	9945	0.55
VB	4000	4000	0	10000	8547	14.53

**Tabel 5.** Hasil *CPU Utilization*

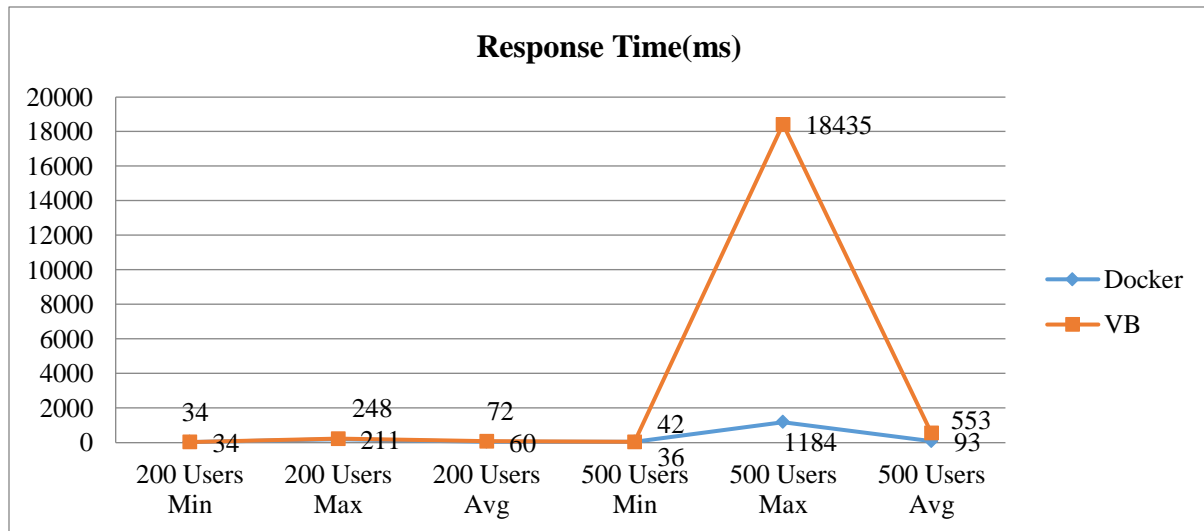
Server		CPU Utilization	
Virtualisasi	Sistem Operasi	Idle	Running (High)
<i>Docker</i>	Ubuntu	0	9.0
VB	Ubuntu	0	52.5
	Windows	14	99

### Pembahasan

Melalui data-data yang didapat melalui performance testing, ada beberapa variabel yang menjadi pengukur performa dari masing-masing virtualisasi. Untuk mengetahui performa dari masing-masing virtualisasi dilakukan peninjauan terhadap hasil yang ditunjukkan oleh masing-masing variabel pada performance test yang telah dilakukan. Variabel-variabel yang ada antara lain adalah sebagai berikut:

#### 1. *Response Time*

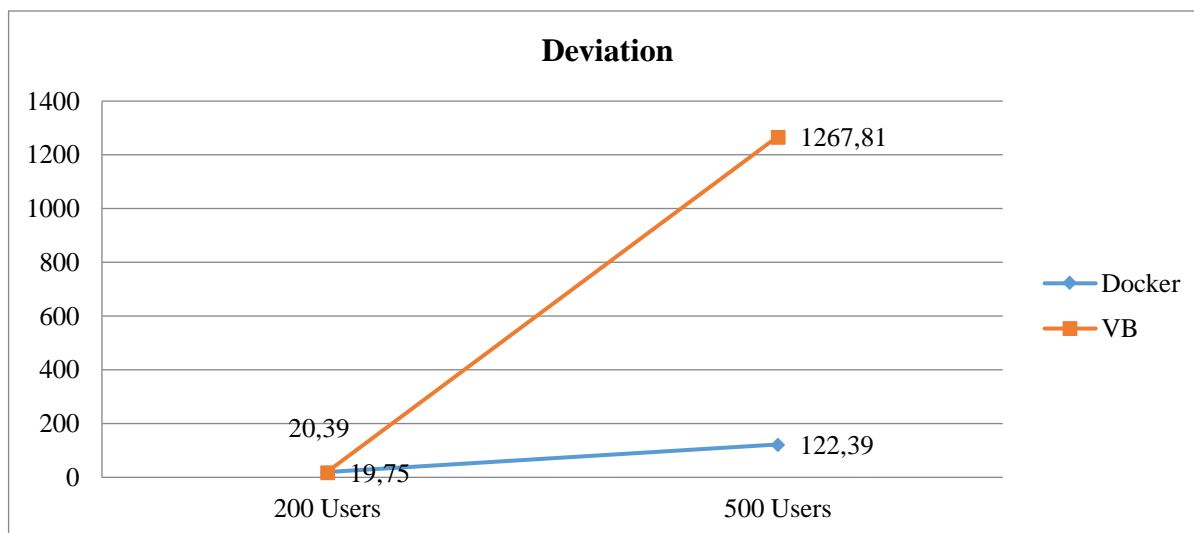
Dari data yang didapatkan, bisa dilihat bahwa pada load 200 user, kedua server memiliki performa yang tidak beda jauh, namun masih bisa dilihat bahwa *docker* memiliki performa yang sedikit lebih bagus, dengan *average* dan *max response time* dibawah VB. Namun pada load 500 user, dapat dilihat bahwa server pada VB kewalahan untuk merespon user tersebut, dengan *max response time* mencapai 18345 ms dan avg 553 ms, sedangkan *docker* meskipun mengalami penurunan performa namun masih lebih stabil dengan avg *response time* hanya 93 ms dan *max response time* 1184 ms, ini jauh lebih baik dari performa VB.



**Gambar 2.** Grafik *Response Time*

## 2. Deviation

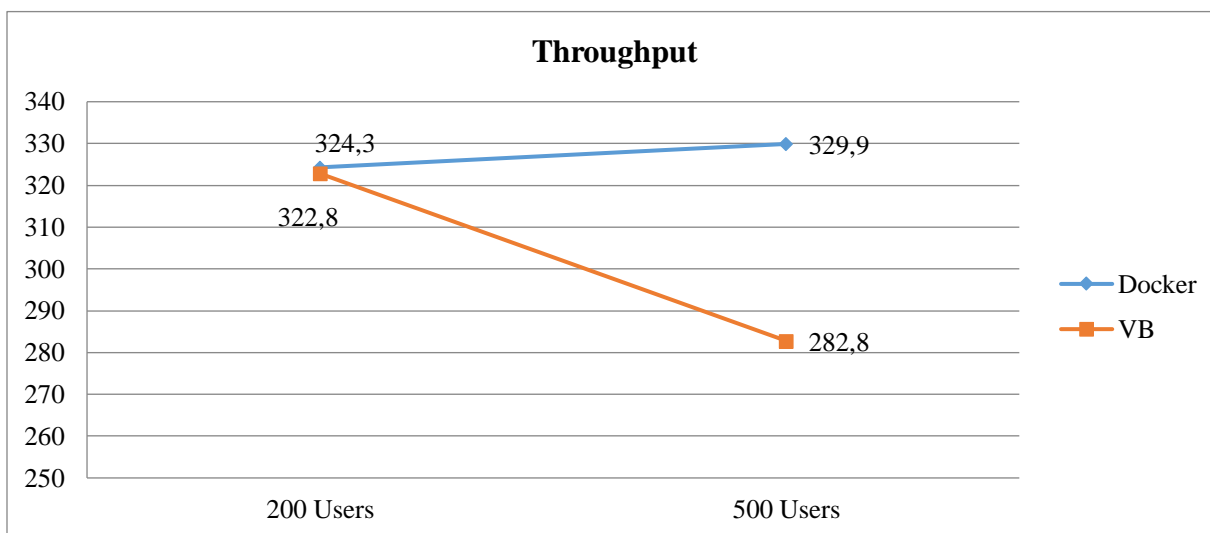
Dari data yang didapatkan, sama halnya dengan response time *docker* dan *vb* tidak memiliki perbedaan yang banyak pada load 200 user dengan *docker* menunjukkan performa yang lebih bagus. Namun pada 500 user, *VB* kewalahan dengan *load* yang ada dan menghasilkan *deviation* sebesar 1276.81 sedangkan *docker* hanya menghasilkan *deviation* sebesar 122.39. pada data *deviation docker* tetap menunjukkan performa yang lebih bagus dari *VB*.



Gambar 3. Grafik Deviation

## 3. Throughput

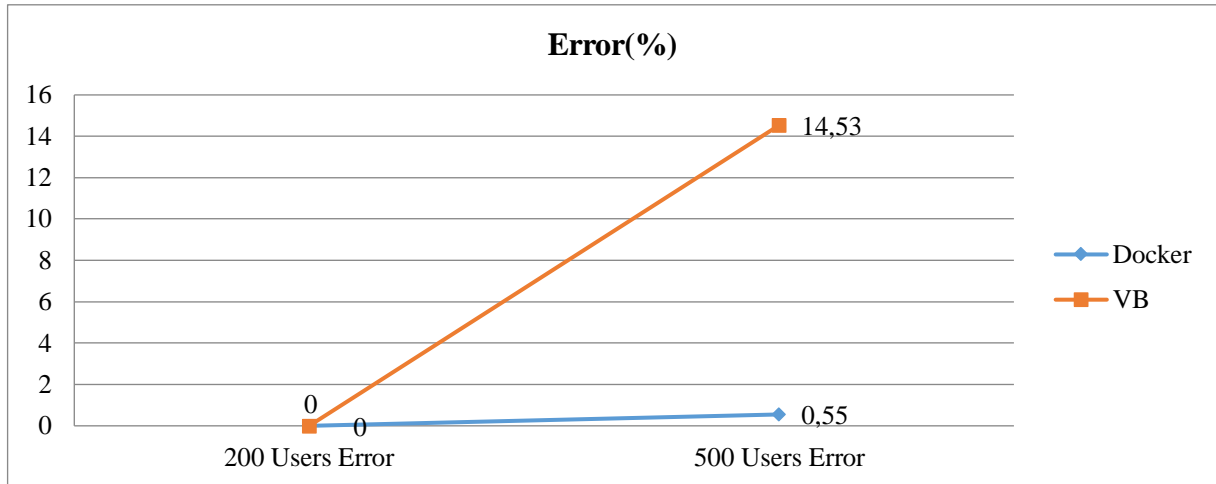
Dari data yang didapatkan dari penelitian, penulis mendapat hasil yang sama dengan kedua variabel sebelumnya, dimana kedua server memiliki performa yang hampir sama pada *load* 200 user dengan *docker* memimpin sedikit didepan *VB*. Sedangkan pada load 500 user *VB* memiliki pengurangan pada nilai *throughput*nya dengan *throughput* yang dihasilkan sebesar 282.8 sedangkan *Docker* malah bertambah menjadi sebesar 329.9. Data *throughput* tetap menunjukkan *docker* berperforma lebih bagus.



Gambar 4. Grafik Throughput

#### 4. Error

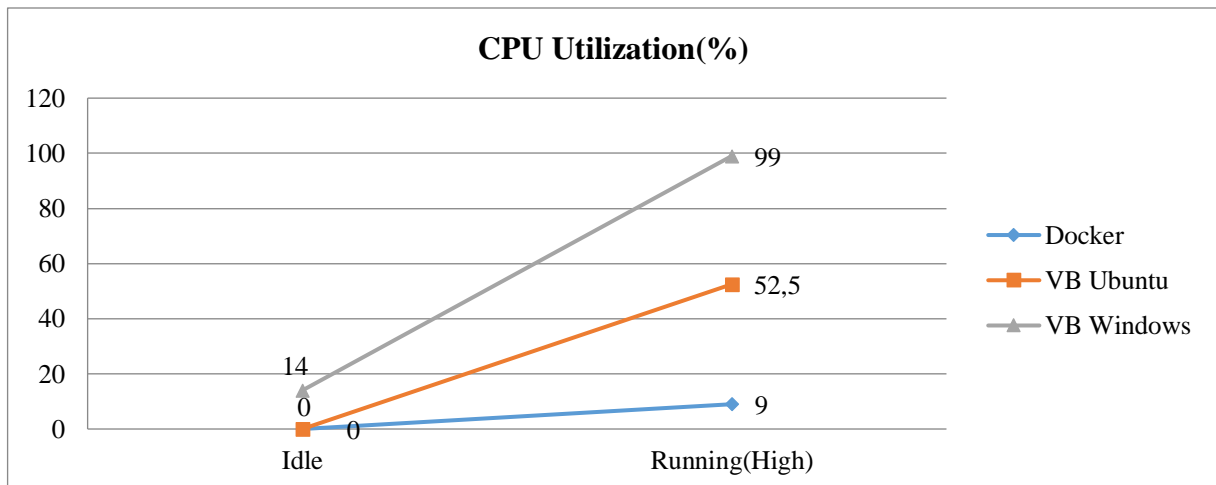
Dengan settingan yang diatur pada loop sebanyak 20 kali, maka *total request* yang dihasilkan pada 200 *user* ada sebanyak 4000 *request*. Kedua server bekerja dengan baik dan tidak menghasilkan *error* pada *load* 200 *user* sehingga seluruh 4000 *request* diselesaikan oleh server. Pada *load* 500 *user*, *total request* yang dihasilkan ada sebanyak 10000, kedua server tidak berhasil menyelesaikan semua *request* yang diminta. Namun *docker* menyelesaikan sebanyak 9945 *request* sehingga hanya memiliki jumlah *error* sebanyak 0.55% sedangkan VB hanya berhasil menyelesaikan *request* sebanyak 8547 dengan *error* sebesar 14.53%. *Docker* menghasilkan performa yang jauh lebih bagus dibandingkan dengan VB.



Gambar 5. Grafik Error

#### 5. CPU Utilization

Pada server *docker*, web server berjalan pada *container* yang dibuat oleh *docker* sehingga hanya berjalan pada satu sistem operasi sedangkan pada server VB, *web server* berjalan pada virtual machine dengan sistem operasi ubuntu yang dibuat pada *instance* yang memiliki sistem operasi windows. Dari data yang didapatkan, CPU Utilization yang dihasilkan oleh *docker* ketika load test dijalankan hanya mencapai 9%, sedangkan pada VB, CPU Utilization pada Ubuntu *virtual machine* mencapai 52.5% dan CPU Utilization pada sistem operasi *instance* mencapai 99%. Ini menunjukkan bahwa arsitektur *docker* terbukti lebih efisien karena memakan *resource* yang lebih sedikit (Herrera-Izquierdo & Grob, 2017).



Gambar 6. Grafik CPU Utilization

Berdasarkan data-data yang didapatkan melalui variabel-variabel yang dihasilkan oleh penelitian didapatkan hasil bahwa virtualisasi berbasis *container* memiliki performa yang lebih bagus dibandingkan virtualisasi berbasis *hypervisor*. Dengan *docker* mewakili virtualisasi berbasis *container* menunjukkan hasil yang jauh lebih baik dibandingkan dengan VB yang mewakili virtualisasi berbasis *hypervisor*. Hal ini senada dengan penelitian yang dilakukan oleh (Potdar et al, 2020) yang melakukan pengujian performa dengan variabel yang berbeda.

Penulis juga mendapatkan bahwa dibawah *load* yang minimal, kedua virtualisasi menunjukkan performa yang mirip dengan *docker* memiliki performa yang sedikit lebih bagus. Namun pada *load* yang besar, virtualisasi berbasis *hypervisor* mulai kewalahan dan menunjukkan performa yang buruk, sedangkan virtualisasi berbasis *container* walau mengalami penurunan performa tetap berfungsi dengan baik dan mampu menyelesaikan request yang diberikan jauh lebih baik dibandingkan virtualisasi berbasis *hypervisor*. Virtualisasi berbasis *container* juga terbukti lebih efisien dibandingkan virtualisasi berbasis *hypervisor* dengan penggunaan sumber daya yang sangat minimal dibandingkan virtualisasi berbasis *hypervisor* seperti yang telah dijelaskan oleh (Kumar & Kurhekar, 2017).

## Kesimpulan

Penelitian yang dilakukan bertujuan untuk mengetahui bagaimana performa dari dua macam virtualisasi, yaitu virtualisasi berbasis *hypervisor* dan virtualisasi berbasis *container*. Beberapa penelitian yang sudah pernah dilakukan sebelumnya menghasilkan hasil yang berbeda, mengingat metode yang digunakan pada masing-masing penelitian berbeda juga. Dengan ada penelitian yang menyebutkan *container* menghasilkan performa yang lebih bagus dan ada yang menyebutkan kedua virtualisasi tidak memiliki perbedaan performa yang signifikan.

Pada pengujian yang dilakukan untuk penelitian ini, dengan menggunakan web server apache sebagai aplikasi tempat pengujian berlangsung menggunakan apache jmeter pada dua jenis virtualisasi yaitu *hypervisor* dan *container* ditemukan bahwa virtualisasi berbasis *container* menunjukkan performa yang lebih bagus dibandingkan *hypervisor*.

Pada *load* 200 *user* kedua *server* menunjukkan performa yang mirip namun ketika dihadapkan dengan *load* yang lebih besar yaitu 500 *user*, *hypervisor* tidak mampu mempertahankan performa seperti yang ditunjukkan pada *load* 200 *user*, sedangkan *container* walaupun mengalami pengurangan performa namun menunjukkan performa yang lebih bagus secara signifikan dibandingkan dengan *hypervisor*.

Ada beberapa hal yang bisa ditambahkan untuk melakukan penelitian lebih lanjut seperti menggunakan mesin fisik untuk melakukan pengujian yang lebih realistis dan tepat karena pengujian ini dilakukan pada *instance cloud* yang terkategori sebagai mesin virtual juga. Selain itu juga bisa ditambahkan variabel-variabel dan aplikasi lain untuk melakukan pengujian terhadap sistem untuk mendapatkan hasil yang lebih komplit.

## Ucapan Terima Kasih

Ucapan terima kasih kepada Program Studi Sistem Informasi, Fakultas Ilmu Komputer Universitas Internasional Batam dan dosen pembimbing serta LP3M UST Yogyakarta yang memberikan dukungan dalam penelitian ini.

## Daftar Pustaka

- Herrera-Izquierdo, L., & Grob, M. (2017). A Performance Evaluation between *Docker Container* and Virtual Machines in Cloud Computing Architectures. *Maskana*, 127–133.
- Kumar, K., & Kurhekar, M. (2017). Economically Efficient Virtualization over Cloud Using



- Docker Containers. Proceedings - 2016 IEEE International Conference on Cloud Computing in Emerging Markets, CCEM 2016*, 95–100. <https://doi.org/10.1109/CCEM.2016.025>.
- Kurniawan, I. N., & Kurniawan, I. F. (2016). Implementasi Virtualisasi Menggunakan Xen Hypervisor. *Jurnal Manajemen Informatika*, 6(1), 36–42.
- Li, Z., Kihl, M., Lu, Q., & Andersson, J. A. (2017). Performance Overhead Comparison between Hypervisor and Container Based Virtualization. *Proceedings - International Conference on Advanced Information Networking and Applications, AINA*, 955–962. <https://doi.org/10.1109/AINA.2017.79>.
- Morabito, R., Kjällman, J., & Komu, M. (2015). Hypervisors vs. Lightweight Virtualization: A performance comparison. *Proceedings - 2015 IEEE International Conference on Cloud Engineering, IC2E 2015*, 386–393. <https://doi.org/10.1109/IC2E.2015.74>.
- Rad, B. B., Bhatti, H. J., & Ahmadi, M. (2017). An Introduction to Docker and Analysis of its Performance. *IJCSNS International Journal of Computer Science and Network Security*, 17(3), 228–235.
- Raho, M., Spyridakis, A., Paolino, M., & Raho, D. (2015). KVM, Xen and Docker: A Performance Analysis for ARM based NFV and Cloud Computing. *Advances in Information, Electronic and Electrical Engineering, AIEEE 2015 - Proceedings of the 2015 IEEE 3rd Workshop*. <https://doi.org/10.1109/AIEEE.2015.7367280>.
- Wang, B., Song, Y., Cui, X., & Cao, J. (2017). Performance Comparison between Hypervisor- and Container-based Virtualizations for Cloud Users. *2017 4th International Conference on Systems and Informatics, ICSAI 2017, 2018-Janua(Icsai)*, 684–689. <https://doi.org/10.1109/ICSAI.2017.8248375>.
- Widarma, A., & Siregar, Y. H. (2019). *Analisis Kinerja Teknologi Virtualisasi Server ( Study Kasus : Universitas Asahan ). Vm*, 688–698.